

О.О. Мазурова¹, Т. О. Гордієнко²

¹Кандидат технічних наук, доцент кафедри Програмної інженерії,
Харківський національний університет радіоелектроніки,
oksana.mazurova@nure.ua, ORSID ID: 0000-0003-3715-3476

²Магістр, напрям «Програмна інженерія»,
Харківський національний університет радіоелектроніки,
trsgdrnk@gmail.com, ORSID ID: 0000-0002-9563-7076

ДОСЛІДЖЕННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ПОШУКУ ОПТИМАЛЬНИХ ШЛЯХІВ В СИСТЕМІ ПРОВЕДЕННЯ МЕТРОМАРАФОНІВ

Робота присвячена дослідженню генетичних алгоритмів на прикладі пошуку оптимальних шляхів для підтримки проведення метромарафонів в системах типу «Transit Challenge» або «Subway Challenge». На основі правил проведення метромарафонів та задачі комівояжера сформульована проблема станцій — пошуку оптимального шляху, який би дозволив відвідати всі станції метрополітену за найкоротший час. На базі теорії графів розроблена математична модель системи метромарафону. Для вирішення проблеми станцій розроблено генетичний алгоритм: обрано спосіб представлення геному, правила мутації популяцій і схрещування геномів. На основі експериментального дослідження генетичного алгоритму підібрано найбільш ефективні параметри та розроблено рекомендації стосовно вирішення проблеми станцій для метромарафонів різної розмірності.

ГЕНЕТИЧНИЙ АЛГОРИТМ, ГРАФ, ЗАДАЧА КОМІВОЯЖЕРА, МАТЕМАТИЧНА МОДЕЛЬ, МЕТРОМАРАФОН, ПОШУК ОПТИМАЛЬНОГО ШЛЯХУ, ПРОБЛЕМА СТАНЦІЙ, СХРЕЩУВАННЯ ГЕНОМІВ.

Работа посвящена исследованию генетических алгоритмов на примере поиска оптимальных путей для поддержки проведения метромарафонов в системах типа «Transit Challenge» или «Subway Challenge». На основе правил проведения метромарафонов и задачи коммивояжера сформулирована проблема станций - поиск оптимального пути, который бы позволил посетить все станции метрополитена за кратчайшее время. На базе теории графов разработана математическая модель системы метромарафона. Для решения проблемы станций разработан генетический алгоритм: выбран способ представления генома, правила мутации популяций и скрещивания геномов.

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ, ГРАФ, ЗАДАЧА КОММИВОЯЖЕРА, МАТЕМАТИЧЕСКАЯ МОДЕЛЬ, МЕТРОМАРАФОН, ПОИСК ОПТИМАЛЬНОГО ПУТИ, ПРОБЛЕМА СТАНЦИЙ, СКРЕЩИВАНИЕ ГЕНОМА.

The work is devoted to the study of genetic algorithms on the example of the search for the best ways to support “Transit challenge” or “Subway Challenge” system. Based on the rules of “Transit challenge” and the tasks of the salesman, the problem of stations was formulated - the search for the best way to visit all subway stations in the shortest possible time. Based on the graph theory, a mathematical model of the metrophone system was developed. To solve the station problem, a genetic algorithm has been developed: the method of genome representation, population mutation and genome crossing rules have been chosen. On the basis of the experimental study of the genetic algorithm the most effective parameters were selected and recommendations on the solution of the station problem for “Transit challenge” of different dimensions were developed.

GENETIC ALGORITHM, GRAPHS, TRAVELING SALESMAN PROBLEM, MATHEMATICAL MODEL, UNDERGROUND CHALLENGE, FINDING THE OPTIMAL PATH, STATION PROBLEM, GENOME CROSSING.

Вступ

Метро — це досить популярний засіб пересування у великих містах. З недавніх пір метро стало визивати навіть певний спортивний інтерес. Так у багатьох країнах світу серед мандрівників та просто зацікавлених у такого роду змаганнях популярним став сценарій використання метро в рамках проведення метромарафонів.

Це змагання називають по-різному: «Transit Challenge» чи «Subway Challenge». В країнах СНГ воно більш відомо як метромарафон. Суть метромарафону проста: відвідати всі станції метрополітену за найкоротший час. Існує навіть сайт rapidtransitchallenge.com, де зафіксовані правила та рекорди цього змагання.

Перед тим як почати змагання учасники заздалегідь планують маршрут та час, який буде затрачено.

Складність цього процесу в тому, що чим більший метрополітен, тим більше різних маршрутів існує. А врахування навіть таких незначних деталей, як інтервали між електропоїздами в різний час, час зупинки на станціях та завантаженість, може дати вигравш у часі під час таких змагань.

В Україні є два великих метрополітени — Харківський та Київський, які мають 30 та 52 станції відповідно. Але в Україні поки що такі змагання не проводилися, відсутні також публікації стосовно результатів побудови відповідних маршрутів. Отже, існує практична задача створення системи «Transit Challenge» для проведення українських метромарафонів, та відповідно наукова проблема побудови найкращого маршруту для них.

1. Аналіз основних досліджень

З метою пошуку шляхів метро краще за все моделюють у вигляді графа, де вершини графу – станції, а перегони – ребра. Методів рішення такого роду задач в теорії графів існує багато. В випадку проведення метромарафону, де потрібно знайти маршрут, який включає всі станції, задача зводиться до задачі комівояжера [4]. Вона має досить високу складність обчислення, якщо знаходити її точний розв'язок. Тому часто використовують евристичні методи, які хоч і не дозволяють знайти оптимальний маршрут, але дають гарний компроміс між оптимальністю маршруту та затратним часом на його пошук.

Деякі з цих евристичних методів намагаються скоротити вже побудований маршрут, видаляючи з маршруту групи ребер та замінюючи їх іншими, що дозволяє отримати новий маршрут. Інші методи комбінують пошук глобальних та локальних розв'язків.

Такі методи як, мурашиний алгоритм, генетичні алгоритми та нейронні мережі використовують ідеї природних процесів та показують досить високу ефективність [1, 2].

2. Постановка задачі

Отже, на основі теорії графів необхідно розробити математичну модель системи метромарафону, яка буде охоплювати всю предметну область разом з існуючими правилами, а також розробити генетичний алгоритм пошуку маршруту для участі в метромарафоні. Також необхідно спланувати та провести експериментальне дослідження алгоритму, виробити рекомендації стосовно параметрів алгоритму та його використання на моделях метромарафону різної розмірності.

3. Розробка математичної моделі системи метромарафону

При моделюванні систему метрополітену доцільно представити у вигляді графів, де вершина – це станція, а дуга – це перехід чи перегон між станціями. Середній час, необхідний для переходу з однієї вершини до іншої, моделюється як вага дуги. Оскільки метрополітен – це дуже налагоджена система, яка працює по чіткому графіку, то такий середній час наближається до фактичного часу здійснення кожного перегону метрополітену.

З пошуком оптимального шляху по всім станціям метрополітену тісно пов'язана проблема комівояжера, так звана (Traveling Salesman Problem, TSP), а також її узагальнена версія Generalized Traveling Salesman Problem (GTSP). В них враховується, що потрібно відвідати всі вершини графа. А в нашому випадку ще й не має ніякого значення, яка лінія та напрямок використовуються для того, щоб відвідати станцію [3].

Якщо представити транспортну мережу у вигляді певного графу, то отримаємо наступну структуру:

- скінченний набір станцій, що позначимо за S .
- скінченний набір ліній метрополітену L , де кожна лінія $L \in L$ – це впорядкований набір станцій.

Припустимо, що кожна лінія працює в обох напрямках і що кожна лінія складається щонайменше з двох станцій. Отже, тепер систему можна змодельовати як орієнтований граф $G = (V, A)$, де V – множина вершин, які зв'язані між собою множиною ребер A . Цей граф має такі властивості:

- для кожної лінії $L \in L$ і всіх станцій s ($s \in L$) завжди є початкова вершина $\vec{v}(s, L) \in V$;
- для кожної лінії $L \in L$ і всіх станцій s ($s \in L$) завжди є кінцева вершина $\bar{v}(s, L) \in V$.

Визначимо множину всіх вершин $s \in S$, що належать станції, як:

$$V(s) = \cup_{L \in L, s \in L} \{ \vec{v}(s, L), \bar{v}(s, L) \}.$$

Позначимо дуги, що відповідають сегментам, які утворюють перегін між двома станціями, як $(\vec{v}(s, L), \bar{v}(s', L))$, де s' наслідуює s для всіх $s \in L$. Аналогічно позначимо дугу $(\bar{v}(s, L), \vec{v}(s'', L))$, де навпаки s наслідуює s'' . Позначимо як R набір дуг, що з'єднує вершини тільки однієї лінії та напрямку.

Для моделювання переходів додамо дуги (u, v) для кожної можливої пари вершин у кожному $V(s)$. Такі дуги дозволять змінювати напрямки.

Кожна дуга $a \in A$ має певну вагу $t(a) > 0$. Якщо a належить до R , то будемо вважати, що це час в дорозі між станціями, а якщо $a \notin R$, то це відповідає часу, необхідному на перехід до іншої станції. Отриманий граф назовемо метрографом (графом метрополітену) від L та S .

На основі такої моделі метромарафону можемо сформулювати проблему станцій та проблему перегонів. В рамках предметної області більш цікава проблема станцій, яка полягає в тому, що потрібно відвідати всі вершини змодельованого графу, саме вона задовольняє умовам, при яких має значення відвідування кожної станції, а не проїзд по всім перегонам на лінії.

Користуючись моделлю метромарафону сформулюємо проблему станцій наступним чином: нехай $G = (V, A)$ є графом метрополітену з вагами дуг $t(a)$, де $t(a) > 0$ та $t(a) \in \mathbb{Q}$, тоді необхідно знайти найдешевший маршрут T , де кожній станції $s \in S$ принаймні один вузол в V_s належить T .

Проблема станцій дуже тісно пов'язана з асиметричним випадком задачі комівояжера і є NP-повною. Математична модель такої задачі буде виглядати таким чином:

$$\min \sum_{a \in A} t_a x_a,$$

де a – певна дуга, що $a \in A$, $t(a)$ – функція розрахунку ваги ребра a , $x(a)$ – значення з множини $\{1, 0\}$, при якому враховується чи належить дане ребро до оптимального маршруту чи ні.

Також за умовами проблеми станцій ми повинні додати до математичної моделі певні обмеження. Обмеження

$$\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = 0 \quad \forall v \in V$$

гарантує, що рішення складається тільки з набору підмаршрутів. Де $\delta^+(v)$ – набір дуг без вершини v , а $\delta^-(v)$ – набір дуг з вершиною v .

Завдяки другому обмеженню

$$\sum_{a \in \delta^+(V_s)} x_a \geq 1 \quad \forall s \in S$$

кожна станція відвідується принаймні один раз в деякому підмаршруті, де S – це множина станцій.

Обмеження

$$\sum_{a \in \delta^+(C)} x_a + \sum_{a \in \delta^-(C)} x_a \geq 2 \quad \forall C \subset V, C$$

$$x_a \in \mathbb{N}_{\geq 0} \quad \forall a \in A$$

гарантує, що кожна станція з'єднана з усіма іншими станціями. Тобто кожен елемент множини станцій повинен мати хоча б по одному зв'язку в кожному з напрямків.

Розглянемо варіанти функції t , яка відповідає за розрахунок ваги кожної з дуг:

– для розрахунку часу на перегоні між двома станціями використовуємо такі залежності:

$$t = K + F, \quad K \in \mathbb{N}_{\geq 0}, \quad F \in \mathbb{N}_{\geq 0},$$

де K – час, який витрачає потяг на подолання перегону між станціями у хвилинух, F – інтервал руху поїздів в певний час доби (у хвилинух);

– для розрахунку на перехід між двома лініями:

$$t = J + F, \quad J \in \mathbb{N}_{\geq 0},$$

де J – це час на перехід між станціями; слід зазначити, що він може відрізнятись в залежності від того, в якому напрямку необхідно перейти зі станції на станцію.

4. Розробка генетичного алгоритму для пошуку оптимального маршруту

Задача комівояжера є транс обчислювальною та відноситься до класу NP-повних задач. Для Харківського метрополітену кількість можливих рішень складає 2.6525286×10^{32} при кількості станцій 30, а для Київського це число набагато більше – 8.0658175×10^{67} на 52 станціях. Якщо уявити, що комп'ютер здатний аналізувати по 1×10^{23} рішення в секунду, то на розрахунок оптимального шляху знадобиться щонайменше 84 роки.

Ідея генетичних алгоритмів вперше була представлена у 1960 році як техніка оптимізації, яка наближає результат до оптимального рішення [5]. Отже ідея використати генетичні алгоритми для рішення задачі комівояжера є цілком перспективною.

Для застосування генетичного алгоритму потрібно описати основні параметри та функції, які буде застосовано в процесі його виконання. До основних

складових алгоритму належать: популяція, особина, геном та хромосома. Після визначення їх представлення потрібно сформулювати основні функції генетичного алгоритму: функцію відбору, пристосованості, схрещування та функції мутації.

Кожне рішення проблеми метромарфону можна представити як вектор P , що є набором всіх станцій в порядку їх слідування в маршруті. Тоді в алгоритмі вектор P буде грати роль геному або, інакше кажучи, набору хромосом $\{P_1 \dots P_n\}$. Але таке рішення приховує декілька неприємних моментів. По-перше, при такому підході ми отримаємо вектори різної довжини, які буде дуже складно схрещувати між собою, а по-друге, багато векторів буде невалідними. В результаті кожен з геномів потрібно буде перевіряти на валідність, що збільшить складність алгоритму.

Кращим рішенням буде взяти на роль геному перестановку множини станцій метрополітену. Для Харківського метрополітену це перестановка 30. Але при такому підході поряд може опинитися пара станцій між якими немає прямого маршруту. Для вирішення цієї проблеми було вирішено перейти до повного графу. Для цього потрібно буде розрахувати найкоротші маршрути між кожною парою станцій. Складність такого розрахунку висока $O(n^3)$, але для Харківського метрополітену це всього лише 27000 варіантів.

Функцією пристосованості має бути цільова функція, яку можна застосувати в ролі порівняльного показника якості. В даній проблемі ним є час, який буде затрачено на подолання даного маршруту.

Нехай $G_i = \{g_{ik}\}$ ($k = \overline{1, n}$) – хромосомі i -ї особини, де g_{ik} значення k -го гена на i -й особині, а n – кількість генів в хромосомі. Тоді можемо сформулювати умову пристосованості хромосоми:

$$f_i = f(G_i) < f_j = f(G_j),$$

а саме: якщо значення функції пристосованості для хромосоми G_i буде меншим за значення функції для хромосоми G_j , то таку хромосому можна вважати більш пристосованою.

Для розробленого алгоритму було обрано наступні функції мутації, які правильно сходяться між собою та дають змогу вибиратися з локальних оптимумів:

– випадкове перетасування випадкового інтервалу генома; ця мутація є швидкою, але не дуже акуратною; вирішено використовувати її на ранніх етапах еволюції, оскільки в фіналі дуже низька ймовірність поліпшення рішення;

– дзеркальне відображення випадкового інтервалу генома; ця мутація буде використовуватися для виходу з локального оптимуму;

– перестановка K пар випадкових значень в геномі місцями; ця досить непогана, але повільна мутація буде ефективно використовуватися ближче до завершення еволюції;

схрещування двох окремих геномів; в цій мутації будемо обирати точку на геномі, розрізатимемо по ній обидва геноми та мінятимемо їх місцями.

В результаті на кожній ітерації ми застосовуємо ті чи інші мутації і формуємо з їх нащадків нове покоління, додаючи до них декілька випадкових маршрутів (схему алгоритму див. на рис. 1).

5. Дослідження генетичного алгоритму

Для дослідження розробленого генетичного алгоритму використано наступні графи:

- граф на основі маршрутної схеми Харківського метрополітену (30 вершин);
- граф на основі маршрутної схеми Київського метрополітену (52 вершини);

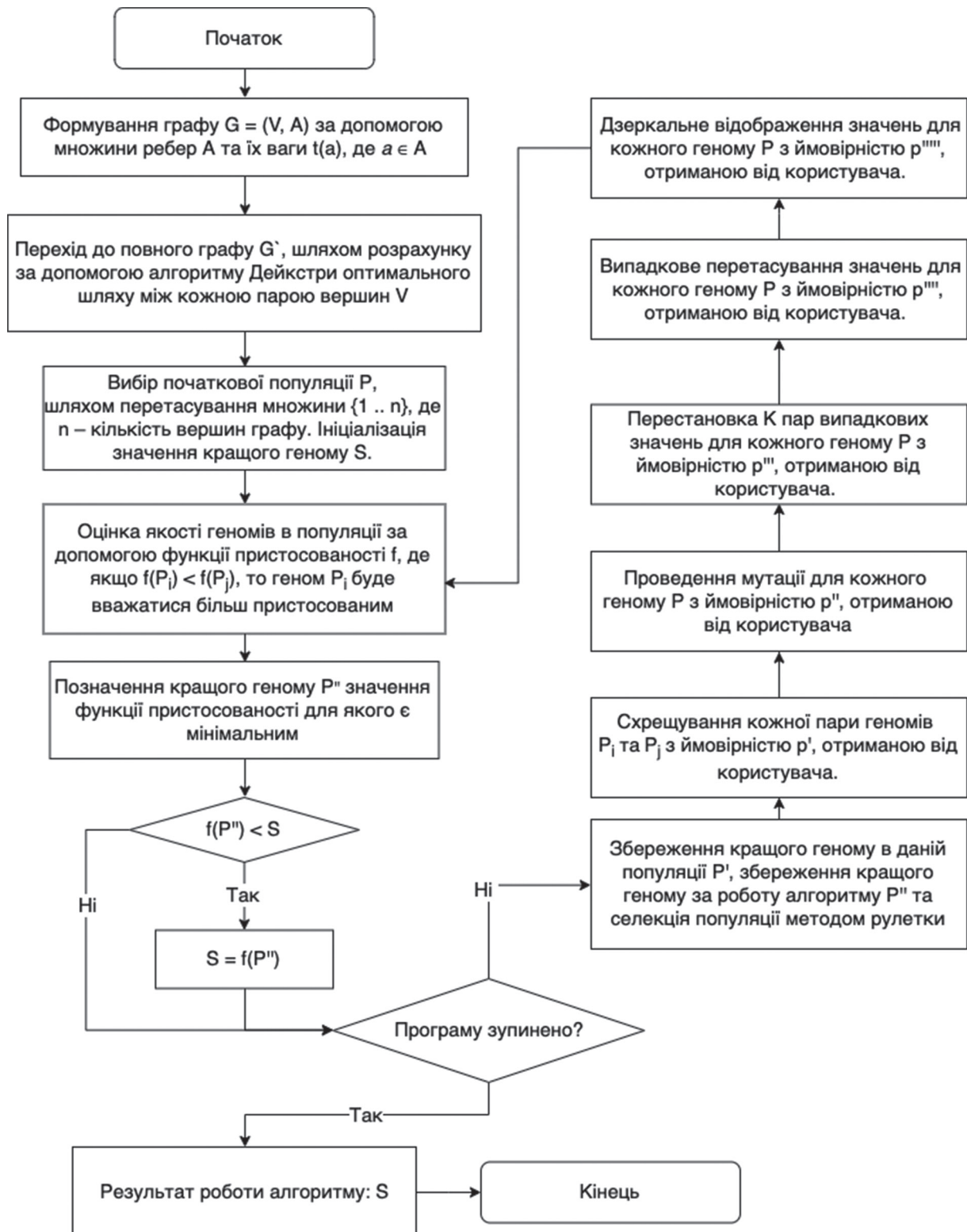


Рис. 1. Схема розробленого генетичного алгоритму

– граф, який імітує транспортну схему з 10 переходами та 50 вершинами, що дозволив дослідити алгоритм в складніших умовах.

Для коректного порівняння ефективності потрібно було обрати час для експериментальної роботи алгоритму. Для цього було проведено кілька пробних запусків алгоритму при різних параметрах на графі Харківського метрополітену. На графіку, наведеному на рисунку 2, точками зображено значення функції пристосованості та час, за який вони були знайдені. З графіку видно, що після 60 секунд алгоритм втрачає темп покращення результату та демонструє стагнацію.

Під час дослідження вирішено змінювати наступні параметри алгоритму:

- кількість особин в популяції – g ;
- ймовірність схрещування геномів – p' ;
- ймовірність проведення мутації – p'' ;
- ймовірність застосування мутації перестановки декількох пар (p'''), випадкового перетасування (p'''') та дзеркального відображення (p'''''').

Алгоритм було протестовано на 5 наборах даних по 3 рази на кожному графі. Кожен тест тривав одну хвилину. В першому наборі даних ймовірність мутації встановлено на 95%. Цей набір має на меті перевірити залежність результату від кількості мутацій. Другий набір даних більш стандартний для такого роду задач

– невелика ймовірність мутацій (5%) та велика схрещування (95%). Третій набір має на меті перевірити роботу з популяцією в 50 особин, тобто більше чим в 2 рази більшою, чим в інших наборах. Четвертий набір базується на зміні ймовірності мутацій і схрещування в процесі роботи програми. Збільшення ймовірності мутації з перестановкою пар та зменшення з випадковим перетасуванням, а також збільшення кількості схрещувань. У п'ятому наборі стратегія зміни параметрів повністю протилежна попередньому пункту. Набори параметрів для проведення експерименту наведено в таблиці 1.

Кращі результати дослідження (вимірюються в хвилину проходження оптимального маршруту) на всіх трьох графах наведено в таблиці 2.

Таблиця 1

Ймовірність/№	1	2	3	4	5
Мутації взагалі	95%	5%	10%	10%	10%
Мутації з перестан. пар	10%	5%	10%	5→30%	30→5%
Мутації з перетасуванням	10%	5%	10%	30→5%	5→30%
Мутації дзерк. відображення	5%	5%	5%	5%	5%
Схрещування	1%	98%	80%	10→80%	80→10%
К-сть особин в популяції, шт	20	20	50	20	20

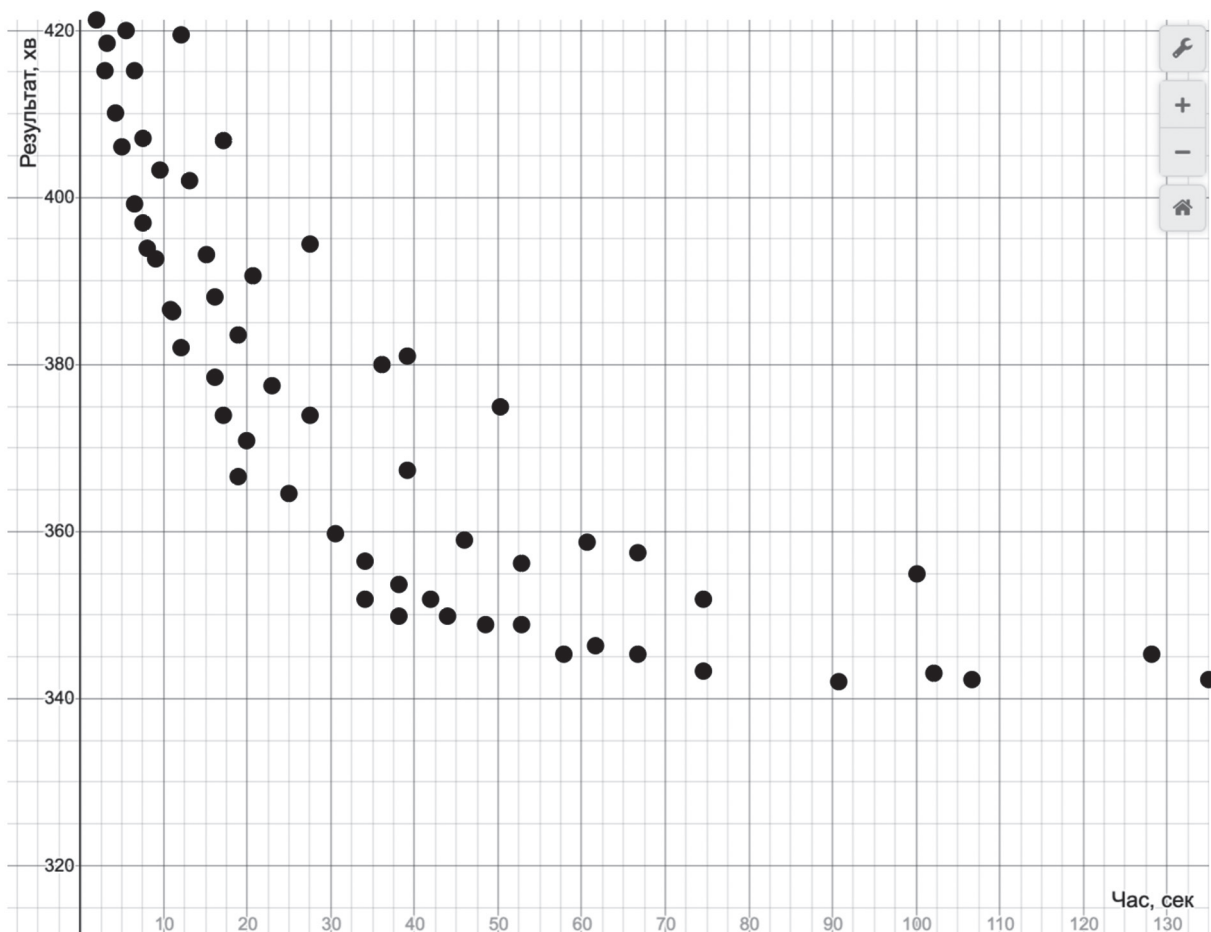


Рис. 1. Схема розробленого генетичного алгоритму

Таблиця 2

Граф	Експеримент номер				
	1	2	3	4	5
Харківський	314	252	262	242	258
Київський	572	415	409	409	421
Штучний	657	430	445	414	433

Аналізуючи на порівнюючи отримані дані, можна зробити висновок, що найоптимальнішою виявився набір параметрів номер 4, як і було прогнозовано. Набір даних №1, де ймовірність мутації була найвищою, показав найгірший результат. Це пояснюється тим, що особини дуже швидко мутували, не встигаючи обмінятися між собою генами.

Другою частиною експерименту є порівняння результату звичайної роботи алгоритму з початковими параметрами та роботи з ручним корегуванням параметрів. Очікувалось невелике зменшення фінального результату при ручному корегуванні параметрів. Тестування було проведено зі штучним графом, що має 50 вершин та 10 переходів. Кращі результати другого етапу наведено в таблиці 3.

Отже, зміна параметрів хоч і не суттєво, але впливає на фінальний результат. В результаті в кожному експерименті був знайдений більш оптимальний маршрут.

Таблиця 3

Вид тестування	Експеримент номер	
	1	2
Без корегування	427 хв.	421 хв.
З корегуванням	410 хв.	409 хв.

Отже, під час вирішенні проблеми метромарафону за допомогою генетичних алгоритмів не рекомендовано задавати високу ймовірність мутацій або змінювати її за межі 10 відсотків. Мутація з випадковою перестановкою краще працює на початку роботи алгоритму. Після швидкої зміни значень варто спробувати зменшити її ймовірність та збільшити ймовірність мутації з перестановкою пар. Якщо значення довго перебуває в локальному оптимумі, варто збільшити вірогідність мутації з дзеркальним відображенням або ймовірність схрещування.

Висновки та перспективи

В роботі на основі теорії графів запропоновано математичну модель системи метромарафону та сформульовано проблему станцій як проблему пошуку оптимального маршруту для метромарафону, який би охопив всі станції метрополітену. Для зниження складності вирішення цієї NP-повною задачі було запропоновано пошук оптимального маршруту за допомогою генетичних алгоритмів.

В розробленому алгоритмі в якості геному було взято перестановку по кількості станцій метрополітену та перейдено до повного графу, сформульовано правила схрещування геномів між собою, що дозволяють вибиратися з локальних оптимумів.

Результати експериментального дослідження дозволили знайти оптимальні значення для параметрів алгоритму, що дозволяє знайти найкращий маршрут в метрополітені за більш-менш реальний час, не застосовуючи повний перебір варіантів.

Список літератури:

- [1] *Murat Albayrak and Novruz Allahverdi*, Development a new mutation operator to solve the traveling salesman problem by aid of genetic algorithms. Expert Systems with Applications, vol. 38, no. 3, 2011 – pp. 1313–1320.
- [2] *Chakroborty, P., and Dwivedi, T.* Optimal route network design for transit system using genetic algorithms. Optimization and Engineering. 34(1), 200 с. — 2003.
- [3] *Dijkstra, E.W.* A note on two problems in connexion with graphs. Numerische Mathematik. 1, 471 с. — 1959.
- [4] *Fan L. and Mumford C. L.* A Metaheuristic Approach to the Urban Transit Routing Problem. Journal of Heuristic. 16, 372 с. — 2010..
- [5] *Ngamchai, S. and Lovell, D.* Optimal time transfer in bus transit route network design using a genetic algorithm, Journal of Transportation Engineering. 129 (5), 510–521 с. — 2003.
- [6] *Baaj, M. H., and Mahmassani, H. S.* An AI-based approach for transit route system planning and design. Journal of Advanced Transportation. 25(2), 287 с. — 1990.

Надійшла до редколегії 4.11.2019